Performance Analysis of cdma2000 Wireless Standard Error Correcting Codes
Md. Ashraf Uddin Bhuiyan and Md. Shazzad Hosain†
Dept. of CSE, The University of Asia Pacific, Dhaka, Bangladesh.
†Dept. of CS, Wayne State University, Detroit, MI, USA.
akash@uap-bd.edu, shazzad@wayne.edu

Abstract
Third Generation Partnership Project-2 proposed cdma2000 standards, which offers two error correcting coding techniques, Convolutional Codes and Turbo Codes. In this paper we analyze the encoding and decoding techniques of the convolutional codes of cdma2000 standards. The performance, in the form of BER (Bit Error Rate) vs. $E_b/N_0$ (bit energy per noise spectral ratio), by varying code rate, frame size, number of decoding iteration of the standardized turbo codes is shown. It is found that if architectural components of turbo codes is changed, not only the BER performance but also the decoding delay and throughput are changed. Results, presented in this paper, facilitate the service provider to trade off among delay, throughput and BER performance as per its requirement.

Keywords: cdma2000, Convolutional Codes, Turbo Codes, decoding iteration, Bit Error Rate

I. INTRODUCTION
Around fifty years ago Shannon demonstrated in his paper [1] that by proper encoding of the information, errors in a noisy channel could be reduced to any desired level. Since then, many efforts were taken to invent such type of codes to reduce the effect of channel noise and interference. Hamming codes is the first error correcting block codes. Convolutional Code follows it and it performs better than block codes. These error correcting codes (ECC), block or convolutional, can reach within only 3 dB to the Shannon limit. Shannon limit for a binary modulation scheme with 1/2 rate, is BER = 0 (several authors take BER = $10^{-5}$ ) for $E_b/N_0 = 0$ dB. Turbo codes (TC) first introduced in [2], and could achieve energy efficiencies within only 0.7 dB of the Shannon’s limit. Now TC is one of the main options for ECC in the third generation mobile standards as cdma2000.
In [3], symbol based turbo codes for cdma2000 standards were proposed which reduces the required memory by 30%. Reference [4] proposed the improved version of Soft Input Viterbi Algorithm (SOVA) for the decoding of cdma2000 turbo codes and the performance was compared with Maximum A Posteriori (MAP) algorithm as decoding algorithm. Unequal power allocation scheme was proposed in [5] which guarantees optimum allocation of power. But the effects of the variation of the architectural component of turbo codes, on its performance, were not explicitly studied in these literatures. In this paper, we investigate encoder and decoder structure of Convolutional code and the performance of turbo codes as an application in cdma2000 standard. We investigate the effect of different input data sizes of the encoder, code rates [6], interleaver design [7], different channel (Additive White Gaussian Noise-AWGN and Rayleigh) models and decoding techniques on the performance (BER vs. $E_b/N_0$) of cdma2000 TC. Using simulation, these effects are shown in the performance curve of TC. For different frame size, the decoding latency are different, which is presented in paper. From these studies one can trade off among latency, throughput and BER performance and can get the optimum structure of turbo codes for one’s specific application.

Section II presents the Convolutional Codes and its performance. Section III gives the structure of classical turbo codes. Section IV shows cdma2000 TC structure and its performance and finally section V draws the conclusion.

II. cdma2000 CONVOLUTIONAL CODE
Many researchers [8], [9] studied on performance of Convolutional codes. Also the distance properties of convolutional codes are analyzed in [10], [11]. In this section we analyze the encoder and decoder structure and performance of cdma2000 Convolutional code.

A. Convolutional Encoder
Convolutional Encoder in cdma2000 standard may be designed as rate of 1/4, 1/3 or 1/2. Rate 1/4 is used for block length of 3048 or smaller size. Rate 1/3 is used for block length between the size of 3048 and 9192. If the block length is greater than 9192, rate 1/2 Convolutional code is used. We study the performance of the Convolutional encoder of 1/2 rate. Analysis of Convolutional Code of other rates will follow it. In rate 1/2 code, one generator function is $g_0 = 753$ (octal) and other generator function is $g_1 = 561$ (octal). The output will be such that, symbol for $g_0$, i.e., $c_0$ will be the first output and symbol for $g_1$, i.e., $c_1$ will be the second output. The rate 1/2 encoder is shown in Fig.1. The
simulated performance of 1/2 rate and 1/4 rate Convolutional codes is shown in Fig. 2 and the BER performance for different constraint length is shown in Fig. 3.

B. Convolutional Decoder
cdma2000 specifies Non-Systematic Convolutional (NSC) encoders with a constraint length \( K = 9 \). Thus the number of states of the encoder or decoder is 256 and is significantly larger than the number of states in the Turbo-encoder (which is equal to 8). Convolutional-decoding, in contrast to Turbo-decoding, is non-iterative. The decoding process of the received codes can be implemented by the well-known Viterbi Algorithm (VA), which detects the maximum likelihood sequence or by the Maximum A Posteriori (MAP) algorithm that detects the maximum likelihood bit. MAP algorithm performs better than VA. Moreover it provides soft information in the decoding decisions. Although there is no interleaver and the block size is much smaller than for Turbo-codes, the architecture of a cdma2000 Convolutional-decoder is also dominated by memory. The I/O memories are, compared to the Turbo-decoder, rather small.

III. CLASSICAL TURBO CODE
Concatenated codes for the search of optimum codes are initially motivated by theoretical research interests. To find codes whose probability of error decreased exponentially while decoding complexity increased only algebraically, Forney [12] arrived at the solution of multilevel coding structure known as concatenated code.
After that a Serially Concatenated Constituents Codes (SCCC) [13] and Parallely Concatenated Constituent Codes (PCCC) [2], [10] were proposed. We concentrate our analysis only on PCCC.

A. TC Encoder

TCs are parallel concatenation of two or more Recursive Systematic Convolutional codes with random interleaver. Fig. 4 and Fig. 5 depict the encoder structure and a simple constituent RSC encoder respectively.

![Fig. 4 Basic Turbo Encoder](image)

![Fig. 5 Turbo RSC Encoder](image)

The interleaver is the critical part of the TC. It permutes the order of the data bits in irregular manner. There are many types of interleaver; e.g., block, pseudorandom and random interleaver. At the encoder the input data stream and two outputs redundancy are serialized into a single TC word. It is a rate 1/3 encoder. If puncturing is implemented to the outputs of the redundant bits, the rate may be increased to 1/2.

B. TC Decoder

The decoder is constituted by two soft-input-soft-output (SISO) decoder with interleaver and de-interleaver. The basic concept of decoding is based on alternately decoding the component codes and passing the extrinsic information, which is a part of the soft output of the SISO decoder, to the next decoding stage. After transmission over a binary symmetric channel, Gaussian or Fading channel, we can calculate the log-likelihood ratio of x conditioned on the output y [14]

\[ L(u / z) = \log \frac{P(x = +1 | y)}{P(x = -1 | y)} \]

where \( P(x = +1 | y) \) is the probability of transmitting +1, given that y is received and \( P(x = -1 | y) \) is the probability of transmitting -1, given that y is received. Ideally the soft-output from the component decoder for the information bit u can be represented by

\[ L(\hat{u}) = L(u) + L_C . y + L_E (\hat{u}) \]

where \( L(u) \) is the a-priori value and is produced by the preceding SISO decoder, \( L_C . y \) is the weighted received systematic channel value and \( L_E (\hat{u}) \) is the extrinsic information. This extrinsic value \( L_E (\hat{u}) \) is passed between the SISO decoder.

IV. cdma2000 Turbo Code

As a forward error correcting coding technique cdma2000 [15] recommends Turbo Codes of variable rate. TC encoder encodes the data, frame quality indicator and two reserve bits. At the end of the encoding, tail sequence are padded with the encoded bits.

A. cdma2000 Turbo Encoder

Turbo encoder are constructed to produce the output at a variable rate of 1/2, 1/3, 1/4 and 1/5. The constituent convolutional code has the code generator of

\[ G(D) = \begin{bmatrix} 1 & g_1(D) & g_2(D) \\ 0 & g_0(D) & g_0(D) \end{bmatrix} \]

Where \( g_1(D) = 1 + D + D^3 \), \( g_2(D) = 1 + D + D^2 + D^3 \), \( g_0(D) = 1 + D^2 + D^3 \). Initially, the states of the constituent encoder registers are set to zero. Puncturing patterns on the encoder output can be found in [15] and they are applied to get variable code rate. The code rate options in a cdma2000 TC are 1/2, 1/3, 1/4 and 1/5. The encoder structure is shown in the Fig. 6.
B. Trellis Termination

Trellis termination technique has the effect on the performance of turbo codes [16]. The encoder generates $6/(\text{code rate})$ tail output symbols after the encoding of the input data. The first $3/(\text{code rate})$ tail output symbol are generated by clocking Constituent Encoder-1 three times while the Constituent Encoder-2 is not clocked. The last $3/(\text{code rate})$ tail output symbols are generated by clocking Constituent Encoder-2 three times while the Constituent Encoder-1 is not clocked. Then the output symbols are punctured and repeated according to the pattern given in [15].

C. Interleaver

The approach of the interleaver are equivalent to write the input bits sequentially into an array, and then the entire sequences are read out from a sequence of addresses [17]-[18]. The interleaving algorithm of cdma2000 is as follows:

1. Turbo interleaver parameter, $n$ is determined, where $K \leq 2^{n+5}$, $K$ is the input bits to the Turbo encoder.
2. $(n+5)$-bit counter is initialized to 0.
3. Extract the $n$ MSBs from the counter and add one to form a new value. Discard all except $n$ LSBs.
4. $n$ bits output are found in a look-up table found in [15] using the five LSBs of the counter.
5. Multiply the values in steps 3 and 4. Discard all except the $n$ LSBs.
6. Five LSBs of the counter are reversed.
7. A tentative output address that has its MSBs equal to the value obtained in step 6 and its LSBs equal to the value obtained in step 5 are taken.
8. Accept the tentative output address as an output address, if it is less than $K$. Otherwise discard it.
9. Increment the counter and repeat steps 3 through 8 until all $K$ interleaver output addresses are obtained.

D. Performance of the cdma2000 TC

Fig. 7 BER performance of cdma2000 TC for two Channel (FS-762, rate-1/2)
In Fig. 7 cdma2000 TC performance curves are shown for two channel model, AWGN and Rayleigh channel model. At each channel model performance for different iteration are shown. It is found that TC performs better with AWGN channel model than with Rayleigh channel model. From curve of Fig. 7, it is found that for 8 decoding iteration and BER of $10^{-3}$, required $E_b/N_0$ is 0.8 dB smaller for AWGN model than Rayleigh model. In Fig. 8 code performance for different frame sizes are shown and it is found that if we increase the frame size from 1530 to 12282, at BER of $10^{-4}$, 0.5 dB can be reduced. In Fig. 9 code performance for different decoding iterations are shown. It is found that if we increase the decoding iteration from 4 to 8, at BER of $10^{-3}$, required $E_b/N_0$ can be reduced by 0.4 dB. In Fig. 10, variable code rate performance is shown. cdma2000 TC has the option to produce code rate of 1/2, 1/3, 1/4 and 1/5 by applying puncturing pattern. It is found from Fig. 10 that if we decrease code rate from 1/2 to 1/5, at BER $10^{-3}$, 0.9 dB improvements can be achieved. But with the decrease of code rate, throughput is also reduced. So there should be a trade-off between BER and code rate. Again from Table I, it is found that with the increase of frame size, latency is also increased. So there should be a trade-off between latency and BER while choosing the frame size.

V. CONCLUSION

cdma2000 standard TC has a variable code rate (1/2, 1/3, 1/4 and 1/5). So its code rate may vary depending on the application. By using Matlab simulation, it is shown how the performance of cdma2000 TC varies with the variation of its architectural component-code rate, length of information bits, number of decoding iteration and the channel model. The analysis of error correcting codes presented in this paper, facilitate the network service provider to select the optimum architecture of turbo codes for their application.

REFERENCES


Table I Delay and BER of TC at $E_b/N_0=1.5$ dB

<table>
<thead>
<tr>
<th>Frame Size</th>
<th>Decoding delay (@ 128 kbps)</th>
<th>BER</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>16 msec</td>
<td>2.4x10^{-5}</td>
</tr>
<tr>
<td>1024</td>
<td>64 msec</td>
<td>9x10^{-3}</td>
</tr>
<tr>
<td>4096</td>
<td>256 msec</td>
<td>1.2x10^{-5}</td>
</tr>
<tr>
<td>16,384</td>
<td>1024 msec</td>
<td>4.3x10^{-6}</td>
</tr>
</tbody>
</table>

Fig. 8 BER performance of cdma2000 TC for different frame size (it-5, rate-1/2, AWGN)

Fig. 9 BER curve of cdma2000 TC for different decoding iteration (FS-762, AWGN, rate-1/2)

Fig. 10 BER performance of cdma2000 TC for different coding rate (FS-762, AWGN, it-5)


